

Activation weight switching.

Public domain. Sean O'Connor 3 September 2021.

The ReLU activation function has 2 states.

$\text{ReLU}(x) = x$ when $x \geq 0$

$\text{ReLU}(x) = 0$ when $x < 0$

In a fully connected ReLU neural network of width n , a neuron in one layer is forward connected to n weights in the next layer. When the input (x) to that particular neuron's ReLU activation function is greater than zero a fixed pattern defined by the forward weights is projected onto the next layer with intensity x . When x is less than zero nothing is projected forward.

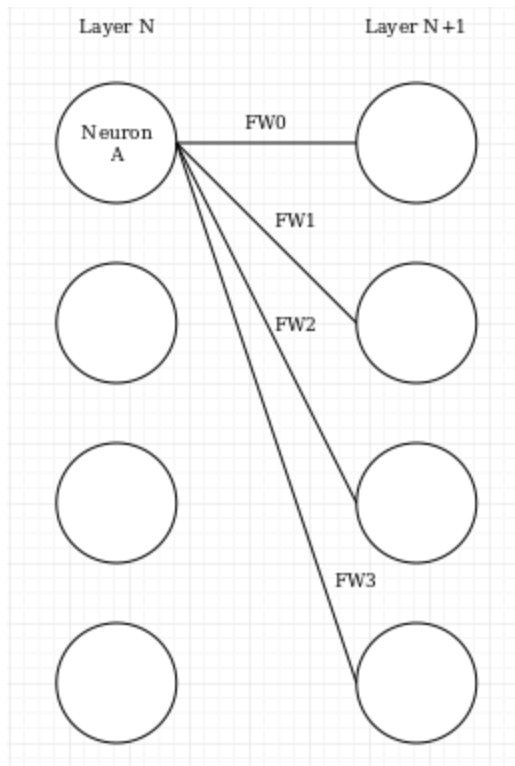


Figure 1. The forward weights of Neuron A in layer N

Approximately 50% of the time nothing gets projected forward. Blocking the flow of information.

If each neuron had an alternative set of forward weights, then when x was less than zero you could project that pattern forward with intensity x rather than blocking. In that case the activation function is always $f(x) = x$, what changes is an alternative set of forward weights is used when $x < 0$.

Then you can have an activation function that switches between 2 possible forward connected weight vectors for each neuron and projects the chosen vector with intensity x onto the next layer.

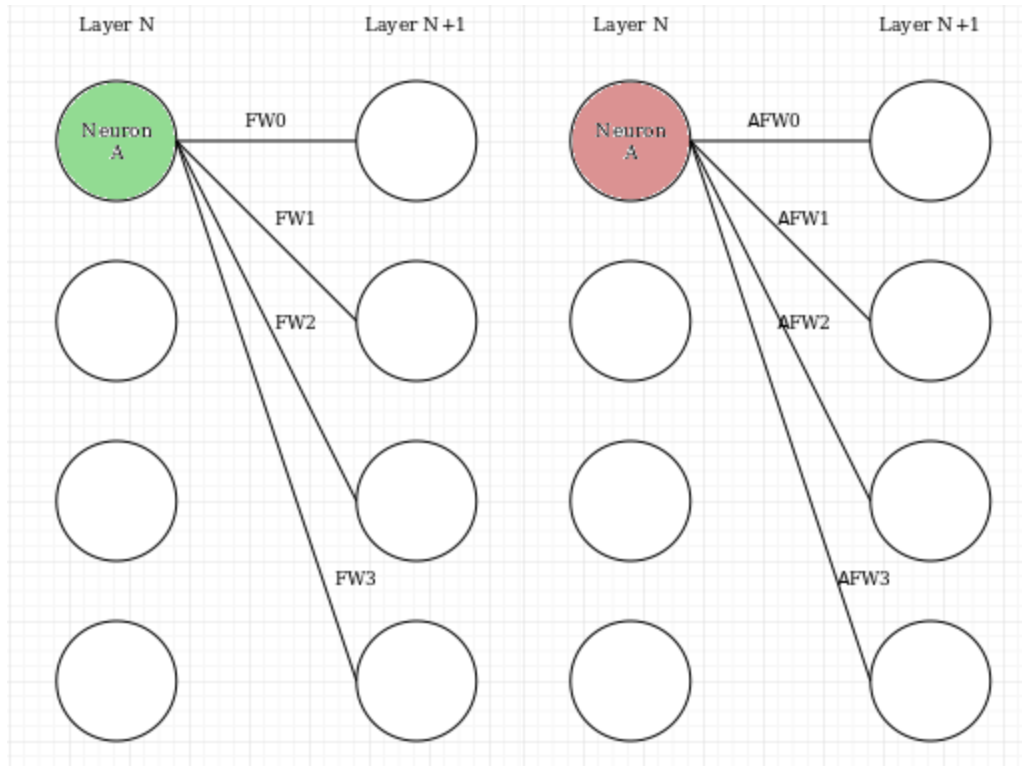


Figure 2. When Neuron A is in the positive activation state ($x \geq 0$, green) one set of forward weights is chosen. When Neuron A is in the negative activation state ($x < 0$, red) a different set of forward weights is chosen. The actual activation function simplifies to $f(x)=x$ with, though, entrained weight switching.

Positive Negative State Weight Switching (PNW)?